the exception is found, then the return association value at the bottom of the association stack is removed. When the thread is completed, the association return value at the bottom of the association stack is returned to the user.

The execution stack is used to optimize association thread performance. It allows thread execution to continue at a specified quanta in the event of a "dead end". This happens, for example, if a match condition has multiple executions based on different field descriptors, and one of the exceptions has an exception to it (an exception to an exception). In this case, execution should continue at the first match conditions' quanta (not the preceding exceptions' quanta), in order to look for the next exception.

When an association thread is started, the user specifies a base set of field descriptors to begin with. As the association thread executes, other field descriptors are invoked by the field descriptor references contained in associated quantas.

The number of behaviors is not limited and may include almost any imaginable logical function. One of the behaviors is the association set. This indicates that the current quantas' association value should be pushed on to the association stack. Another behavior is the qualifier set. This indicates that additional key data should be iconized as specified in the referenced field descriptor and a subsequent lookup should be attempted. The possible effect of the next association is not known until it is found. Versions of the association set (M) and the qualifier set (QM) are explained with respect to figures 1-6. Another behavior is the test set. This set contains an addition field for a score. As a thread is

processed the association with the highest score is maintained. Any association that does not have a higher score is ignored. Another behavior is an exclusion set. This indicates that the quanta represents an exception, so the return association value at the bottom of the association stack is removed. Another behavior is the continuation set. This indicates that processing should continue.

FIG. 18 is an example of a behavioral operation. Assume that a user wants to find the keys 370 with the associations 372. An associative memory with every entry could be created, however another alternative exists with behavioral sets. The keys 370 and associations 372 could be represented by the quantas 374. Note that the "x" indicates a don't care. The first quanta 376 indicates that the range of keys 5550-555F are potential matches. We know this because the behavior type (flag) is "A-Q". The Q behavior tells us to investigate further using field descriptor "2". The field descriptors 378 are listed below. The next quanta 380 shows that upon further investigation the key "555F" is excluded, but any of the other keys in the range will return the association "A". Quantas 382, 384, 386 are used to define when the association "B" is returned. Note that field descriptor "2" 388 indicates an offset of "0" bytes or start at the zero byte and investigate to the first byte. The mask 390 indicates "FFFF" which means all bits in the two bytes are to be processed. A "0" bit would indicate a don't care bit. While more complex searches may be created using the system the example shows the power to reduces the number of quantas that have to be created. In this example the number of quantas was reduced from

twenty-nine to six and this is just part of the power of the behavioral operation system.

FIG. 19 is a flow chart of the steps used in a method of behavioral operation of a data document in accordance with one embodiment of the invention. The process starts, step 400, by matching a pattern of data at step 402. Next a behavior set associated with the pattern is determined at step 404. At step 406 an action indicted by the behavioral set is performed which ends the process at step 408. In one embodiment the step of matching a pattern of data includes determining an icon for the pattern. Next an associative lookup using the icon is performed to determine if a match exists. In one embodiment, the action performed may include storing an association and acquiring an information connected to the association. An association usually points to a location in a store where additional information about the match may be found. For instance, the pattern might be a customer's name. The association would point to a location in the store where the customer's address may be found. In another embodiment, the action may be determining a new field of data to be examined.

FIG. 20 is a flow chart of the steps used in a method of behavioral operation of a data document in accordance with one embodiment of the invention. The process starts, step 420, by scanning an input data to find a match at step 422. When a match is found, a behavioral set associated with the match is determined at step 424. When the behavioral set is an association set at step 426, an association in the match is used to acquire a desired information which ends the process at step 428. In one embodiment, when the behavioral set is a qualifier set,